

**SIMULADOR DE HABITAÇÃO INTELIGENTE**

Programação Orientada a Objetos

Ano Letivo 2023/2024

Pedro Paiva - 2021134625

Tomás Laranjeira - 2021135060

Licenciatura de Engenharia Informática

Índice

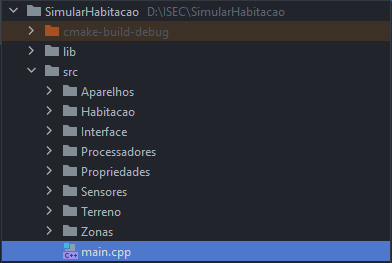
Introdução

Este relatório aborda o desenvolvimento de um simulador em C++ para uma habitação controlada por domótica. O simulador engloba várias zonas da habitação, cada uma com propriedades como temperatura e luz, monitorizadas por sensores. Estes sensores fornecem dados a processadores de regras que, por sua vez, determinam ações com base nas leituras obtidas. Além disso, as zonas possuem aparelhos que afetam as suas propriedades e respondem a comandos dos processadores de regras.

Apesar da aparente simplicidade dos comandos, o simulador é, essencialmente, um conjunto de objetos interligados com uma interface de utilizador. Destaca-se, igualmente, a forma como o simulador aborda a passagem do tempo, utilizando unidades abstratas denominadas "instantes", que avançam conforme os comandos emitidos pelo utilizador.

Este relatório fornecerá uma visão abrangente da arquitetura, implementação e funcionamento deste simulador domótico em C++.

Organização do código



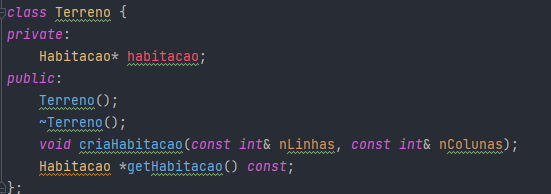
**\lib**: Neste diretoria, agrupámos todos os ficheiros relacionados à biblioteca fornecida pelo professor.

**\src:** Esta é a principal pasta do projeto, onde segmentámos o código em subdiretorias específicas.

* **main.cpp**: Funciona como o ponto de entrada central do projeto.
* **\Terreno**: Dedicada à implementação da classe "Terreno".
* **\Habitacao**: Contém os ficheiros relacionados com a classe "Habitacao".
* **\Interface**: Reservada para os ficheiros da classe "Interface".
* **\Zonas**: Estrutura os elementos da classe "Zona".
* **\Aparelhos**: Contém os ficheiros da classe principal "Aparelho" e as suas subclasses (Aquecedor,Refrigerador, Lampada, ... ).
* **\Processadores**: Designada para os ficheiros das classes principais "Processador" e “Regra” e as suas subclasses (EntreRegra, ForaRegra , ...).
* **\Propriedades**: Agrupa os ficheiros relacionados à classe principal "Propriedade" e as suas subclasses Temperatura, Som, ... .
* **\Sensores**: Reservada para a classe “Sensor” e todas as suas subclasses, tais como Temperatura, Movimento, Som, Fumo, entre outras.

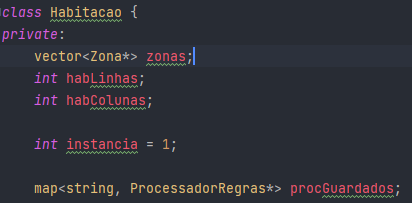
Esta organização visa otimizar a gestão do código, facilitando a colaboração entre membros da equipa e contribuindo para a manutenção e expansão eficientes do projeto.

Classe Terreno



A classe "Terreno" representa uma parcela de terreno no contexto do projeto. Nela, é armazenado um ponteiro para um objeto da classe Habitacao. A sua função principal é permitir a criação de uma habitação neste “terreno”, indicando o número de linhas e colunas desejado. Em resumo, esta classe atua como uma entidade que encapsula e gere a habitação associada, facilitando a manipulação e interação com essa habitação no âmbito do sistema.

Classe HABITACAO

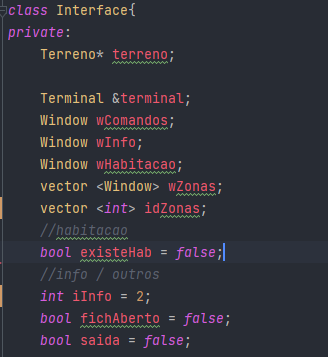


A classe Habitacao é fundamental no projeto, sendo responsável por gerir uma habitação, que por sua vez é composta por diferentes zonas. Aqui estão os principais pontos:

* Zonas: A classe mantém um vetor de zonas, cada uma representa uma área específica na habitação. Essas zonas podem ser manipuladas e geridas pela instância da "Habitacao".
* Processadores de Regras Guardados: A classe mantém um mapa associando nomes a processadores de regras. Este mapa permite guardar e recuperar processadores de regras específicos associados à habitação.
* Operações com Zonas: Fornece métodos para adicionar e remover zonas, bem como obter informações sobre as zonas existentes na habitação.

Resumidamente, a classe "Habitacao" atua como uma entidade central para gerir a configuração, as zonas e os processadores de regras associados.

Classe Interface



A classe "Interface"trata da interação do utilizador com o sistema. Alguns pontos principais:

* Terreno: Mantém uma ligação com a classe "Terreno", permitindo interação com a habitação e consequentemente com as zonas.
* Janelas (Windows): Servem para exibir informações, comandos, detalhes da habitação e zonas.
* Processamento de Comandos: Oferece métodos para processar e executar comandos fornecidos pelo utilizador.
* Construção e Manipulação da Habitação e Zonas: Inclui métodos para construir a habitação, avançar no tempo, criar zonas, limpar zonas e realizar outras operações relacionadas à habitação.
* Verificações: Realiza varias verificações para garantir o melhor funcionamento possível.

Esta classe atua como uma interface entre o utilizador e o sistema, permitindo-lhes interagir e controlar as diferentes funcionalidades do projeto.

Classe Zona

Asda

Classe Propriedade

Ssad

Classe Sensor

Asda

Classe Aparelho

Asda

Classe ProcessadorRegras

asda

Classe Regra

Dsadadsa

DIFICULDADES SENTIDAS AO LONGO DO PROJETO

Arranjar um nome + curto para o titulo

Falar da biblioteca, por ser uma ferramenta nnova e nnc termos usado.

Falar da inclusão circular

Falar da dificuldade em fazer testes para encontrar bugs